The instant rejection is again respectfully traversed because Minear in view of Mason, with or without Gennaro, do not disclose or suggest all elements of all pending claims, for at least the following reasons.

First, consider the Minear disclosure. Minear discloses a firewall and a system and method for regulating the flow of messages through the firewall which has a network protocol stack including an Internet Protocol (IP) layer. An incoming message from the network (i.e., a "datagram") to the firewall comes into the firewall's IP layer (i.e., "the kernel") where it is examined to determine if it is encrypted. (Minear, Fig. 2) If the incoming message is not encrypted, the message is passed up the stack to the application level where a proxy exists. If the message is encrypted, it is first decrypted at the IP layer after which it is passed up the stack to the proxy in the application layer. Decrypting the message is accomplished by executing a process in the IP layer. (Minear, Abstract).

Messages coming in to Minear's firewall (a node) from another node in the network which includes that firewall as one of the network nodes are described as messages conforming to a protocol, namely the Internet Protocol Security (IPSEC) protocol (e.g., col. 1, lines 51-52.) The two main components of IPSEC security are data encryption associated with the payload of the message and sender authentication associated with the header of the message (e.g., col. 1, lines 62-63). The Office Action, in its "Response to Arguments" section, (pg. 3, bottom or pg. 4, bottom) interprets an IPSEC-conformed message in Minear as both Applicants' claimed "input" and Applicants' claimed "message." Applicants respectfully disagree, because this is not an appropriate interpretation of Minear, nor is this an appropriate application of Minear against Applicants' claims.

Consider, for example, claim 1:

> In a node operative within a network of a plurality of nodes, a method for performing cryptographic-related functions, comprising:
>
> executing an application program in a user space at the node;
>
> receiving an input requiring cryptographic-related processing;
>
> generating a message in the node via the application program based on the input, the message being the same as one of a predefined set of messages stored in the node and being processed by one of a plurality of cryptographic processing components located in a kernel space within the node, each one of said messages being associated with a respective one of said cryptographic-related functions;
>
> transmitting the message to one of a socket handler and a call handler in kernel space at the node to obtain a transmitted message;
>
> forwarding the transmitted message to a request handler at the node which generates a function call to the cryptographic processing component appropriate for the transmitted message; and
>
> performing the cryptographic-related processing by the cryptographic processing component appropriate for the transmitted message.

Minear does not disclose or suggest at least Applicants' recited generating step. The Office Action reads the IPSEC messages in Minear as being equivalent to Applicants' recited "input" that requires cryptographic-related processing. (Office Action, pg 6, referring to Minear, col. 5, lines 37-45 and col. 6, lines 13-27). With Minear's IPSEC-conformed messages characterized in that manner in the Office Action, Minear's firewall (e.g., firewall 18 in Fig. 1 in Minear) is necessarily viewed in the Office Action as being equivalent to Applicants' recited "node." Moreover, with Minear's IPSEC-conforming messages being so characterized in the Office Action, those messages cannot also be

simultaneously equivalent to messages generated within the node by Applicants' message generating step. [2]

Notably, Applicants recite the language: "generating a message in the node…" (emphasis added) The context of claim 1 is a method that is performed within a single node, as clearly recited in the preamble of claim 1. Thus, it is in that particular node where the claimed "message" is generated. This language therefore cuts-off any association between Minear's IPSEC-conformed message that is being sent over its network, e.g., via Internet 16 to firewall 18 (Minear, Fig. 1), and Applicants' message which is generated in the recited node. Minear's IPSEC-conformed message is generated from some place outside of its firewall node; the message may be received by its firewall, and processed within its firewall, but it is not generated in its firewall. This is a substantive difference between the teachings of Minear and Applicants' recited generating step. In light of this difference, consider the various passages applied in the Office Action to Applicants' generating step:

> To use IPSEC, you must create a security association (SA) for each destination IP address. In one embodiment, each SA contains the following information:
> Security Parameters Index (SPI)--The index used to find a SA on receipt of an IPSEC datagram.
> Destination IP address--The address used to find the SA and trigger use of IPSEC processing on output.
> The peer SPI--The SPI value to put on a IPSEC datagram on output.
> The peer IP address--The destination IP address to be put into the packet header if IPSEC Tunnel mode is used.
> The Encryption Security Payload (ESP) algorithm to be used.
> The ESP key to used for decryption of input datagrams.
> The ESP key to used for encryption of output datagrams.
> The authentication (AH) algorithm to be used.
> The AH key to be used for validation of input packets.

[2] This stands to reason: IPSEC is applied to a packet for cryptographic purposes. That packet is transferred from one node in a network to another node, e.g., via the Internet. An IPSEC-conformed message is not equivalent to a pre-defined message that is transmitted only within a node, e.g., from application layer to kernel layer. **Certainly, there is no need to encrypt (via IPSEC) a message within a node that is being transmitted only within that node.**

> The AH key to be used for generation of the authentication data for output datagrams. The combination of a given Security Parameter Index and Destination IP address uniquely identifies a particular "Security Association." In one embodiment, the sending firewall uses the sending userid and Destination Address to select an appropriate Security Association (and hence SPI value). The receiving firewall uses the combination of SPI value and Source address to obtain the appropriate Security Association.

(Minear, col. 4, lines 8-36, cited in the Office Action pg 6)[3] The above section relates solely to how IPSEC is used which is related solely to a message coming into Minear's firewall (into the node) from the network. It doesn't relate to generating a message "in the node" as claimed. Consider the next-cited section:

> This means that all communications between two IP addresses will be protected because all interfirewall communication must pass through the IP layer. IPSEC takes the standard Internet packet and converts it into a carrier packet. The carrier packet is designed to do two things: to conceal the contents of the original packet (encryption) and to provide a mechanism by which the receiving firewall can verify the source of the packet (authentication). In one embodiment of the present invention, each IPSEC carrier packet includes both an authentication header used to authenticate the sending machine and an encapsulated payload containing encrypted data.

(Minear, col. 3, line 58 - col. 4, line 3, cited in Office Action, pg 6) Again, the above section relates solely to how IPSEC is used which is related solely to a message coming into Minear's firewall from the network. It doesn't relate to generating a message "in the node" as claimed. Consider the next-cited section:

> Messages coming from unprotected network 16 are reviewed and a determination is made as to whether execution of an authentication and identification protocol is warranted. In contrast to previous systems, system 10 also performs this same determination on IPSEC-encrypted messages. If desired, the same authentication and identification can be made on messages to be transferred from workstation 20 to unprotected network 16. FIG. 2 illustrates one way of authenticating both encrypted and unencrypted messages in a system such as system 10.
>
> In the system of FIG. 2 a network protocol stack 40 includes a physical layer 42, an Internet protocol (IP) layer 44, a Transport layer 46 and an application layer 48. Such a protocol stack exists, for instance on application level gateway firewall 18 of FIG. 1. An application executing in application layer 48 can communicate to an application

[3] The Office Action (pg 6) cited *col. 3*, lines 8-36 in Minear, which starts in mid-sentence, includes the irrelevant Brief Description of Drawings, and ends in mid-sentence; thus, Office Action reference to col. 3 appears to be an error, where the citation was probably intended to be *col. 4*, lines 8-36, as shown above.

executing on another system by preparing a message and transmitting it through one of the existing transport services executing on transport layer 46.

(Minear, col. 5, lines 36-55, cited in Office Action, pg 6) The first paragraph in this section again discusses how IPSEC is used which is related to a message coming into Minear's firewall from either the network or from its associated workstation. It doesn't relate to generating a message "in the node" as claimed.

The second paragraph in this section discusses a protocol stack, shown in Minear, Fig. 2, and how an applications layer 48 in one firewall can communicate with another applications layer 48 in another firewall via transport layer 46. All of this has to do with transmission over the network between firewalls and is irrelevant to generating a message in the node. Consider the next three sections, cited on pg. 6 in the Office Action:

> (I) A message arriving from external network 16 is examined at IP layer 44 and an SADB is queried to determine if the source address and SPI are associated with an SA. In the embodiment shown in FIG. 2, an SADB Master copy 52 is maintained in persistent memory at application layer 48 while a copy 54 of SADB is maintained in volatile memory within the kernel. (Minear, col. 6, lines 3-8)
>
> (II) In the embodiment shown in FIG. 2, a working copy 54 of the Security Association Database consisting of all currently active SA's is kept resident in memory for ready access by IP layer processing as datagrams are received and transmitted. In addition, a working master copy 52 of the SADB is maintained in a file in nonvolatile memory. During system startup and initialization processing the content of all of the required SA's in master SADB 52 is added to the working copy 54 stored in kernel memory. (Minear, col. 6, lines 33-40)
>
> (III) In one embodiment, a flag which accompanies the message as it is sent from IP layer 44 to proxy 50 indicates whether the incoming message was or was not encrypted. In another embodiment, proxy 50 accesses Security Association Database 54 (the table in the kernel can be queried via an SADB routing socket (PF-SADB)) to determine whether or not a security association exists for a given peer. The SADB socket is much like a routing socket found in the stock BSD 4.4 kernel (protocol family PF-ROUTE) except that PF-SADB sockets are used to maintain the Security Association Database (SADB) instead of the routing table. Because the private keys used for encryption, decryption, and keyed authentication are stored in this table, access must be strictly prohibited and allowed to only administrators and key management daemons. Care must be taken when allowing user-level daemons access to /dev/mem or /dev/kmem as well, since the keys

are stored in kernel memory and could be exposed with some creative hacking. Minear, col. 7, lines 23-40)

These three Minear sections, identified above for convenience purposes as sections (I), (II) and (III) are cited collectively in the Office Action against a portion of the generating step which is now currently amended to: "*…the message being the same as one of a predefined set of messages stored in the node and being processed by one of a plurality of cryptographic processing components located in a kernel space within the node, each one of said messages being associated with a respective one of said cryptographic-related functions*." (italics added) These sections do not disclose or suggest this portion of the generating step for the following reasons.

Section (I) discusses the message arriving from the external network, which reinforces the fact that such message in Minear cannot be the same as one of a predefined set of messages previously stored in the node. Whatever message is being received as input to the firewall is certainly not a message that was stored in Minear's SADB. (As noted below, only "SA's" are stored in the SADB anyway, and certainly more than, or other than, only an SA is being received as input to Minear's firewall.)

Section (II) merely refers to two databases, one in the kernel (54) and the other in the application layer (52) where "all currently active SA's" are kept resident in memory. An "SA" is, by Minear definition (col. 4, lines 8-28), a Security Association. Thus, the Minear databases may contain these SA's, but these are not equivalent to, nor can they be read on, "a predefined set of messages stored in the node" where the recited "message" is the same as one of the "predefined set of messages." The message sent from IP layer to proxy 50 is not the same as any of the SA's stored in the SADB.

Section (III) discusses two embodiments. In one embodiment a flag accompanies a "message" sent from IP layer 44 to proxy 50 indicating if the "incoming message" (input) is encrypted or not. However, that message sent from IP layer to proxy 50 including such flag is not the same as any of the SA's stored in the SADB. But Applicants' claim calls for a message in the node being the same as one of a predefined set of messages stored in the node. Therefore, the first embodiment discussed in section (III) does not read on claim 1.

The second embodiment of section (III) discusses proxy 50 in user space which accesses SADB 54 in kernel space to determine if a security association (SA) exists for a "given peer." As previously discussed in a prior response, a "given peer" is referred to in Minear as relating to **OUTPUT**, not input. For example, "The peer SPI - The SPI value to put on a IPSEC datagram on *output*. The peer IP address - The *destination* IP address to be put into the packet header if IPSEC Tunnel mode is used." (Minear, column 4, lines 16-19; emphasis added) Therefore, this second embodiment in section (III) teaches that something may be generated by proxy 50 in user space to determine if an SA exists based on an output. Accordingly, this second embodiment of this section also cannot read on Applicants' recited generating step which is "based on an input."

In summary, all of the above-quoted sections of Minear, which were applied in the Office Action against the message generating step of claim 1, taken individually or collectively, do not disclose or suggest the message generating step of claim 1 which is:

"generating a message in the node via the application program based on the input, the message being the same as one of a predefined set of messages previously stored in the node before receiving the input and being processed by one of a plurality of cryptographic processing components located in a kernel space within the node, each one of said messages being associated with a respective one of said cryptographic-related functions."

Reiterating, (1) the Minear "message" cited in the Office Action is not generated "in the node" but, rather, is a datagram received by the node, wherefore (2) that Minear message is not equivalent to Applicants' one of a predefined set of messages stored in the node, and (3) the Minear SADB database stores SA's and does not store the same as whatever is being transmitted to, or transmitted internal to, Minear's firewall, for example, between kernel and user layers which, therefore, teaches away from Applicants' generating step.

Since Applicants' message generating step in claim 1 is not disclosed or suggested by Minear, at least because Applicants' recited message is not shown in Minear, the additional transmitting, forwarding and performing steps of claim 1 are also not disclosed or suggested at least because they rely on the recited message from the generating step, such message not being taught in Minear. Therefore, for any or all of these reasons, claim 1 itself is not disclosed or suggested by Minear. Mason, cited to teach a call handler, does not cure this deficiency in Minear. Gennaro, cited to allegedly show a technology of PKAI (public key authentication infrastructure), and not cited against claim 1 anyway, does not cure this deficiency of Minear. Accordingly, the 35 U.S.C. § 103(a) rejection of claim 1 as being un-patentable over Minear and further in view of Mason should be withdrawn and the claim allowed.

The other independent claims 5, 9, 14 and 22 have each been amended to include similar language to that discussed above with respect to claim 1. Thus the other independent claims are allowable for reasons similar to those given above for claim 1.All pending dependent claims are allowable for reasons based on their respective dependencies from allowable base claims.

CONCLUSION

All rejections in the Office Action have been addressed.[4] In view of the foregoing remarks, reconsideration and allowance of the pending claims are respectfully requested.

To the extent necessary, a petition for an extension of time under 37 C.F.R. § 1.136 is hereby made. Please charge any shortage in fees due in connection with the filing of this paper, including extension of time fees, to Deposit Account 07-2347 and please credit any excess fees to such deposit account.

Respectfully submitted,

By: 
Joel Wall
Reg. No. 25,648

Date: May 12, 2008

Verizon
Patent Management Group
1515 Courthouse Road, Suite 500
Arlington, VA 22201-2909
Tel: 703.351.3586
CUSTOMER NO. 25,537

[4] As Applicant's remarks with respect to the Examiner's rejections are sufficient to overcome these rejections, Applicant's silence as to assertions by the Examiner in the Office Action or certain requirements that may be applicable to such rejections (e.g., whether a reference constitutes prior art, motivation to combine references, assertions as to dependent claims, etc.) is not a concession by Applicant that such assertions are accurate or such requirements have been met, and Applicant reserves the right to analyze and dispute such assertions/requirements in the future.